



Méthode Agile de Gestion de Projets



Crystal

Elaboré par le groupe 5 :

Belghiss Sori Semeke : C116168

Mahamadou Barry Amadou Cissé : C12517

Sous la tutelle de :

Dr Fatimetou mint Mohamed Saleck

Contenu de ce rapport

1	INTRODUCTION	3
2	Historique de Crystal.....	4
3	Les valeurs partagées.....	4
3.1	Communication.....	4
3.2	Releases fréquentes	5
3.3	Souplesse	5
4	processus de Crystal	5
4.1	Spécifications.....	5
4.2	Conception et planning	5
4.3	Itérations.....	6
5	Avantages et Inconvénients.....	7
5.1	Avantages.....	7
5.2	Inconvénients.....	7
6	Famille Crystal	7
6.1	Principes communs des méthodes Crystal	8
6.1.1	Livraisons fréquentes	9
6.1.2	Un processus d'amélioration continue.....	9
6.1.3	Communication accrue entre les membres de l'équipe.....	9
6.1.4	Mise en confiance de l'équipe	10
6.1.5	La concentration des développeurs.....	10
6.1.6	Accès facilité à des experts	10
6.1.7	Environnement technique.....	10
6.2	Crystal Clear « transparent »	11
6.2.1	Fonctionnement de Crystal Clear	11
6.2.2	Cycle de développement.....	11
6.2.2.1	Le projet.....	12
6.2.2.2	Livraison	12
6.2.2.3	Itérations	12
6.2.2.4	Journées	12
6.2.2.5	Intégration.....	13
6.2.2.6	Episodes	13
7	Scrum	13
7.1	Vocabulaire de Scrum.....	13

8	Comparaison entre Scrum et Crystal	14
9	Conclusion	15
10	Références.....	16

1 INTRODUCTION

Les méthodes agiles se sont imposées par leur manière de penser les projets : management des équipes, alignement permanent du contenu aux objectifs du client, test en continu ,elle reposent sur des cycles de développement itératifs et adaptatifs, dans lesquels une application est produite et testée en continu.

Si l'on demande à un informaticien ou à un responsable de projet de citer une méthode agile de gestion de projet, il y a toutes les chances pour qu'il réponde « Scrum ». S'il s'agit de loin de la méthode la plus répandue et là plus en vue actuellement, c'est loin d'être la seule. Plusieurs méthodes agiles, regroupées au sein d'une même famille appelée « Crystal », sont bien moins connues, et pourtant elles présentent suffisamment d'avantages pour que l'on s'y intéresse. Faisons donc un peu le tour de la méthode agile Crystal et ses 7 caractéristiques.

Crystal apparaît dans le développement agile de projets, entre autres avec le modèle du « scrum », mais aussi avec d'autres méthodes. L'approche Crystal n'est toutefois pas une seule et unique méthode, mais d'un ensemble de différentes démarches qui sont reliées entre elles.

2 HISTORIQUE DE CRYSTAL

Au début des années 1990, Alistair Cockburn initia le développement d'une méthode, qui deviendra plus tard le développement Crystal. Cockburn devait alors mettre au point pour IBM Consulting une méthode servant au développement orienté objet. Cockburn dut faire face alors à un problème de taille : à l'époque, il n'existait pas encore des possibilités de comparaison ; donc, il analysait autant de projets que possible et menait des interviews avec les participants.

Cockburn vint à la conclusion que chaque projet est unique et n'est pas comparable à d'autres projets. Il exclut ainsi les règles uniformes qui ne pouvaient pas satisfaire l'individualité de chaque projet. En lieu et place, il misa sur la communication régulière des équipes de projet qu'il jugeait essentielles. Et il retint quelques dimensions et caractéristiques qui constituèrent Crystal. De 1991 à 1999, les résultats de Cockburn tinrent aussi lieu de norme internationale.

3 LES VALEURS PARTAGEES

Les valeurs partagées par l'équipe Crystal sont :

- Communication
- Releases fréquentes
- Souplesse

3.1 Communication

Alistair Cockburn écrit : « Software development is a cooperative game, in which the participants help each other in reaching the end of the game – the delivery of the software »

Faisant partie des caractéristiques des méthodes agiles, la communication est un facteur crucial dans une équipe qui travaille avec la méthode Crystal.

Afin de faciliter la communication l'équipe entière est de préférence regroupée dans une pièce unique et limitée en nombre (6 ou 7 personnes). Le but du regroupement de l'équipe en un seul lieu est de limiter les sources de distraction des développeurs. La collaboration produit un travail de meilleure qualité, les schémas de modélisation sont généralement faits en groupe sur un tableau blanc, aussi accentuer la collaboration avec les clients / utilisateurs afin de produire un répondant aux attentes du client et les utilisateurs.

3.2 Releases fréquentes

Le principe du manifeste Agile, des parties fonctionnelles du produit doit être soumis au client dans des périodicités bien définies pour négocier les changements. Cela permet d'adopter une procédure formalisée de déploiement.

3.3 Souplesse

Crystal est souple tant au niveau des procédures à suivre que des normes à utiliser (comme les normes de codage). Les normes de codage et de nommage sont très peu contraignantes voire inexistantes pour certains points.

L'équipe est relativement libre de s'organiser à sa guise : parmi les développeurs par exemple, on peut distinguer deux types d'approche. Certains commenceront par prendre une feuille de papier et un crayon pour essayer de schématiser la manière dont ils vont construire et organiser leur code avant de démarrer réellement la programmation. D'autres au contraire fonceront tête baissée dans l'écriture du code ce qui implique pour eux de consacrer plus de temps par la suite au refactoring du code.

Les méthodes agiles Crystal ne sont pas centrées sur les processus mais plutôt sur une équipe de développement qui comprend des personnes aux compétences et aux talents très différents.

4 PROCESSUS DE CRYSTAL

Le processus Crystal se décline de la façon suivante :

- Spécifications
- Conception et planning
- Itérations

4.1 Spécifications

La spécialisation consiste à observer les utilisateurs dans leur travail pour mieux connaître leurs besoins et leur environnement. Ensuite, les différents cas d'utilisation sont classés par ordre de priorité en collaboration avec les utilisateurs, ce qui permet de savoir quelles fonctionnalités ont le plus de valeur et doivent être développées en premier.

4.2 Conception et planning

Une ébauche de conception est réalisée au tout début du projet, cela inclut les choix des technologies à utiliser et implique une ébauche d'architecture.

le planning consiste à prévoir vers quelles dates les itérations vont se suivre, il est recommandé de définir des itérations d'une longueur de 2 à 3 mois, chacune produisant un produit à livrer fonctionnel.

4.3 Itérations

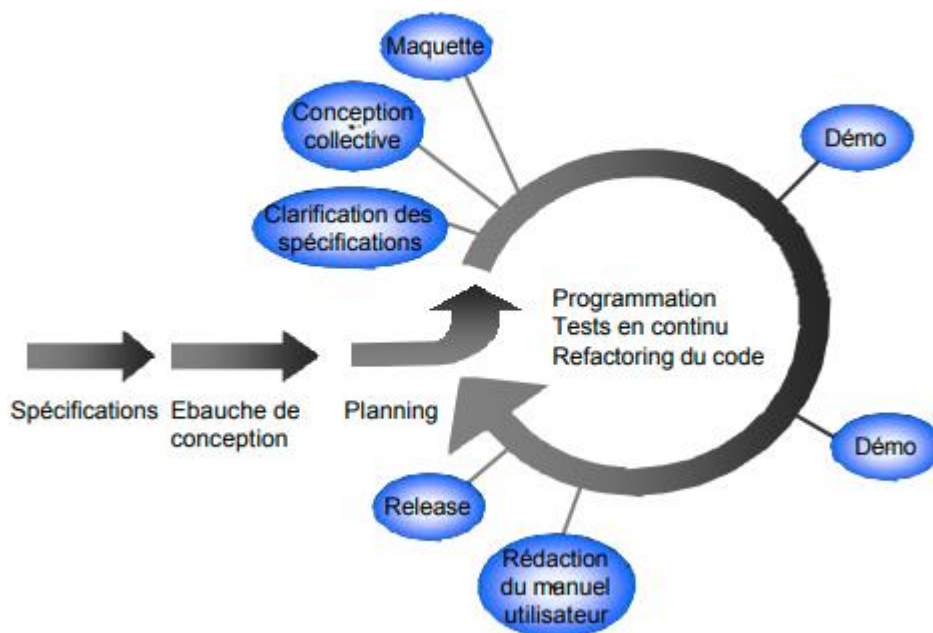
Les itérations, c'est au cours de cette phase que se fait la réalisation proprement dite de l'application, en suivant un ordre de phase.

Le premier travail consiste à clarifier et préciser les spécifications au cours de discussions plutôt informelles entre utilisateurs et développeurs.

Ensuite, les développeurs font tous ensemble un travail de conception pour déterminer la manière dont ils vont mettre en place les objets de base sur lesquels va s'appuyer le système.

Les tests sont omniprésents dans le processus de développement : l'architecture est conçue de manière à pouvoir supporter des tests régressifs et des tests automatiques.

Deux ou trois fois par itérations, des démonstrations sont faites aux clients / utilisateurs ce qui permet de toujours bien recadrer le projet en conformité avec les exigences du client. L'écriture du manuel utilisateur est effectué soit juste avant la livraison d'une release, soit dès le début de l'itération ce qui force les développeurs à définir plus précisément dès le début ce vers quoi ils s'engagent. L'itération se termine par l'intégration et la mise en production de l'application qui est testée une fois de plus avant d'être livrée au client.



5 AVANTAGES ET INCONVENIENTS

Crystal présente tous les avantages des méthodes agiles : flexibilité par rapport au changement, rapidité, livraisons fréquentes, etc. Elle convient tout à fait pour des petites structures (taille inférieure à 6 personnes), mais ce qui fait son efficacité dans les projets de petite taille cause son inadéquation pour des projets plus importants.

5.1 Avantages

- Les composants à haut risque et très importants sont livrés en premier
- Communication efficace de l'équipe et cela facilite l'apprentissage entre les membres de l'équipe les uns des autres
- Cette méthodologie peut être ajustée selon le type de projet et la taille de l'équipe

5.2 Inconvénients

- La planification et le développement ne dépendent pas des exigences, la traçabilité est donc un problème dans Crystal.

6 FAMILLE CRYSTAL

Cockburn a développé des différentes méthodes à partir de la méthodes Crystal pour s'aligner avec des équipes de tailles différentes qui nécessitent des solutions différentes pour résoudre des problèmes divers, un code de couleur est utilisé pour identifier le « poids » de la méthode agile à utiliser en fonction du projet.

Les différentes couleurs utilisées sont :

- Transparent (« Clear »),
- Jaune,
- Orange,
- Rouge,
- Marron,
- Diamant
- Saphir.

Ce qui avec une variante supplémentaire nous donne les méthodes Crystal Clear, Crystal Yellow, Crystal Orange, Crystal Orange Web, Crystal Red, Crystal Maroon, Crystal Diamond et Crystal Sapphire. Plus le projet à gérer est important, et plus la couleur de la méthode est foncée.

On comprend donc que la méthode de management agile Crystal Clear est particulièrement adaptée à des petits projets alors que Crystal Yellow et Orange conviendront à des projets de taille moyenne.

Si par contre le projet comporte une mission critique ou la vie humaine sera exposé au risque alors la méthode Crystal Diamond ou Crystal Sapphire peuvent être utilisées.

	Clear	Yellow	Orange	Red	Maroon
Life (L)	L6	L20	L40	L80	L200
Essential Money (E)	E6	E20	E40	E80	E200
Discretionary Money (D)	D6	D20	D40	D80	D200
Comfort (C)	C6	C20	C40	C80	C200
	1-6	7-20	21-40	41-80	81-200

6.1 Principes communs des méthodes Crystal

S'il existe différentes méthodes dans la famille Crystal adaptées à différents cas de figure et types de projets, toutes partagent sept points communs :

- Livraisons fréquentes,
- Un processus d'amélioration continue,
- Une communication accrue entre les membres de l'équipe,
- La mise en confiance de l'équipe,
- La priorité à la concentration des développeurs,
- L'accès facilité à des experts
- Un environnement technique disposant de tests automatisés et d'outils permettant d'effectuer de la gestion de configuration et des intégrations fréquentes.

6.1.1 Livraisons fréquentes

Les livraisons fréquentes font parties des pratiques de la plupart des méthodes agiles. L'équipe de développement sélectionne les fonctionnalités incluses dans chaque livraison et crée les tests nécessaires. Avec les méthodes agiles Crystal, les livraisons peuvent être plus ou moins rapprochées, toutes les semaines, ou tous les trimestres, en fonction de la taille et de la longueur du projet.

Grâce à des livraisons fréquentes, les membres de l'équipe sont en mesure de détecter plus rapidement les anomalies. Plus une anomalie est détectée tôt dans un projet, et moins elle coûte cher à corriger. De même, si le client s'aperçoit finalement que l'application développée ne correspond pas exactement à ce qu'il voulait, les changements pourront être pris en compte plus facilement que s'il fallait attendre la fin du projet. Contrairement à d'autres méthodes agiles, il est possible qu'une livraison soit composée de plusieurs itérations de développement.

6.1.2 Un processus d'amélioration continue

Le processus d'amélioration continue a pour objectif de permettre à l'équipe de développement de quitter temporairement ses tâches habituelles pour se pencher sur les problèmes qui ont pu être rencontrés. L'objectif est d'analyser les problèmes, d'identifier les causes et de mettre en place les solutions pour qu'ils ne se reproduisent plus, de tirer les leçons de l'expérience acquise. C'est l'occasion également d'optimiser les processus du projet afin que l'équipe soit encore plus efficace d'une itération à l'autre.

Dans les méthodes agiles Crystal, des réunions de réflexion sur l'amélioration continue du projet sont organisées toutes les deux semaines. C'est l'occasion pour l'équipe projet de faire le tour de ce qui a bien fonctionné et de ce qui a moins bien fonctionné afin d'identifier les axes d'amélioration.

6.1.3 Communication accrue entre les membres de l'équipe

La communication entre les membres de l'équipe projet est vivement encouragée. Pour ce faire, l'équipe entière est de préférence regroupée dans une pièce unique. Si l'équipe est trop importante, au-delà de huit personnes, des cloisonnements peuvent être mis en place de façon à maintenir la communication, tout en limitant les sources de distraction et les nuisances. Le but du regroupement de l'équipe en un seul lieu est de limiter les sources de distraction des développeurs. S'ils ont une question à poser ou une demande à faire, ils n'ont pas à se déplacer, à changer de pièce, ils rompent moins longtemps leur concentration et sont plus efficaces.

Les informations circulent plus facilement entre les membres de l'équipe, les questions trouvent des réponses plus rapidement. De plus, en étant proches les uns des autres, les développeurs assistent à toutes les conversations et peuvent ainsi partager leurs expériences et trouver de nouvelles idées. Ce type d'organisation permet également de libérer la créativité de l'équipe projet.

6.1.4 Mise en confiance de l'équipe

Les méthodes agiles Crystal mettent en avant la confiance entre les membres de l'équipe projet. Aucune suggestion, aucune idée ne doit être écartée, tout le monde peut s'exprimer sans avoir la crainte de se voir ridiculisé. Chacun doit avoir suffisamment confiance dans les autres membres de l'équipe pour se sentir libre d'exposer ses idées et ses objections. La confiance est un élément central des méthodes agiles Crystal.

6.1.5 La concentration des développeurs

La concentration dans les méthodes agiles Crystal concerne deux domaines :

- Les tâches attribuées individuellement
- La direction suivie par le projet dans son ensemble.

Afin de permettre aux membres de l'équipe de se concentrer sur les tâches qui leur ont été attribuées, toutes les sources de distractions doivent être proscrites, ou plus raisonnablement limitées le plus possible : interruptions pour des discussions, réunions, questions posées hors contexte, conversations téléphoniques.

Toute interruption de la concentration d'une personne sur sa tâche en cours peut être grandement préjudiciable, car elle aura besoin d'un certain temps pour revenir à son niveau de concentration initial. Il y a donc une perte de temps et de productivité évidente. Pour éviter cela, l'équipe projet devra être mise à l'abri autant que faire se peut de toutes ces distractions.

Ainsi, les méthodes Crystal définissent deux règles simples :

- Un développeur doit travailler sans interruption durant une période de deux heures (on s'isole, on coupe le téléphone et tout moyen de communication susceptible d'interrompre la tâche en cours).
- Un développeur ne doit pas travailler moins de deux jours complets sur un projet avant d'en changer si nécessaire (en-dessous de deux jours, il n'a pas le temps de s'approprier le projet et d'atteindre son efficacité maximale).

6.1.6 Accès facilité à des experts

Le ou les experts techniques et métier associés au projet doivent se rendre disponibles pour répondre à toute question pouvant se poser l'équipe de développement. Ils doivent être totalement impliqués dans le projet, et facilement accessibles. Au moindre doute énoncé par un développeur concernant telle ou telle fonctionnalité, l'expert doit être en mesure de lui répondre dans les plus brefs délais, et de façon la plus précise possible. Il est courant que l'expert ne soit pas disponible 100% du temps pour le projet. Dans ce cas, il devra être possible d'organiser avec lui au moins une réunion hebdomadaire de deux heures, et de le joindre par téléphone à n'importe quel moment.

6.1.7 Environnement technique

Enfin, le dernier point commun entre toutes les méthodes agiles Crystal concerne l'environnement technique utilisé par l'équipe de développement. Tous les outils nécessaires

à l'intégration continue des développements (tests et déploiements automatisés, remontées d'alertes en cas de détection d'anomalie...) doivent être mis en place dès le début du projet. C'est à cette condition que les anomalies pourront être détectées rapidement, et corrigées.

Plus une anomalie est détectée tôt, et moins le coût de sa correction sera élevé. L'utilisation de ces outils et d'un gestionnaire de code source performant (SVN et Git sont les plus répandus actuellement) permettra de détecter les anomalies mais également de revenir à une version précédente du code source si les modifications nécessaires sont trop lourdes pour être faites rapidement. Un outil de gestion des configurations va également permettre d'effectuer des livraisons avec des paramétrages différents, adaptés à l'environnement ciblé (la configuration sera différente suivant que la livraison a lieu sur un serveur de tests, d'intégration, de recette ou de production).

6.2 Crystal Clear « transparent »

Le Crystal Clear est une méthode Agile cyclique et itérative. C'est une méthode particulière car elle ambience le projet mais ne qualifie pas vraiment comment faire vos projets structurellement.

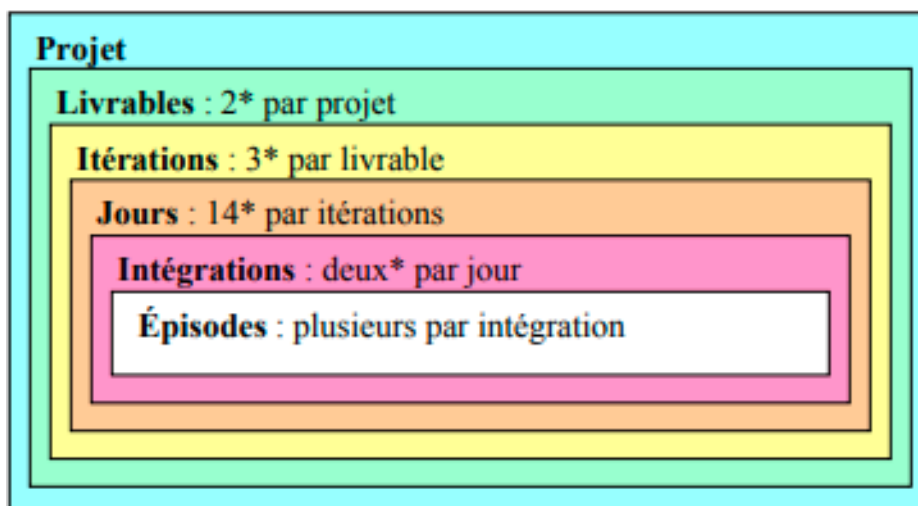
6.2.1 Fonctionnement de Crystal Clear

Crystal Clear si on se permet de le dire est la méthode agile rebelle, elle ne suit pas une conduite méthodologique elle procure un environnement de travail favorisant la concentration et la communication transparente entre les membres.

Le travail se fait par itération et l'équipe est capable d'apprendre de son propre fonctionnement et de l'adapter de façon optimale ce qui leur crée une façon de faire propre à elle non formalisées et qui respecte la philosophie agile sans même le dénoncer.

6.2.2 Cycle de développement

Comme les autres approches agiles, le modèle itératif incrémentale est repris. Crystal Clear est composé de cycles imbriqués. La méthode proposée nous permet de schématiser les cycles et leurs répétitions comme une série d'imbrications.



6.2.2.1 Le projet

Le projet est l'unité de base et peut être de durée variable. Il comporte une étape de planification initiale, au moins deux cycles de livraisons et une étape de fermeture.

La planification initiale : sélectionner l'équipe principale en charge du projet, déterminer les conventions et la méthodologie à suivre pour la réalisation du projet.

Cycles de livraisons : Correspond aux différentes versions de du produit livré.

Étape de fermeture : la clôture du projet, par une dernière réflexion sur son déroulement afin d'en dresser le bilan formateur.

6.2.2.2 Livraison

Le cycle de livraison peut durer d'une semaine à trois mois. Il se termine par la mise en place d'une version du produit fonctionnel pour l'utilisateur. Un cycle de livraison comporte une révision de la planification, une ou plusieurs itérations, une étape de livraison et une séance de réflexion

Révision de la planification : commence à partir la second livraison, l'équipe est en mesure d'évaluer le travail par rapport à la livraison précédentes afin de réviser pour celle qui débute. Il peut en résulter de remplacer l'équipe, changer la portée ou l'échéancier du projet ou trouver une solution plus créatrice pour arriver dans les délais avec l'équipe en place.

Les itérations : Construction du produit pour une nouvelle livraison à l'utilisateur. Un produit fonctionnel et utile pour une véritable production, il est possible de déployer des versions intermédiaires à des petits groupes d'utilisateurs, afin d'obtenir le feedback nécessaire à la progression du projet.

Séance de réflexion : C'est la fin du cycle, on révisé d'une part, une réflexion sur le produit en cours de réalisation et d'autre part une réflexion sur les conventions mises en place qui composent la méthode.

6.2.2.3 Itérations

Ce cycle prend en charge une portion des travaux d'une livraison. Elle peut durer d'une semaine à trois mois.

Pour une itération, la planification affecte les tâches aux membres de l'équipe. Les développeurs sélectionnent les tâches qu'ils estiment pouvoir réaliser au cours de l'itération. Lorsque chacun sait ce qu'il a à faire, le travail journalier débute.

À la fin de l'itération, les membres de l'équipe participent à une courte séance de réflexion et de célébration. L'objectif de cette activité est surtout de marquer un temps d'arrêt, afin de prévenir l'épuisement. Cette pause reconfortante évite d'épuiser les gens.

6.2.2.4 Journées

Chaque journée comprise dans une itération comporte une rencontre de l'équipe de développement et aucune ou plusieurs intégrations. Les rencontres journalières effectuent le

contrôle et le suivi des réalisations. Consulter la section sur les techniques pour plus de détails.

6.2.2.5 Intégration

Une intégration comporte plusieurs épisodes, se termine par une compilation et des tests. Dépendamment du projet, une intégration peut se produire à toutes les demi-heures ou aux quelques jours. Un environnement technique bien adapté, permet d'accélérer sa réalisation. Une série de tests sont réalisés à chaque intégration, afin de valider immédiatement le bon fonctionnement du produit avec les nouveaux ajouts.

6.2.2.6 Episodes

Un épisode est la portion qui inclut le design, le codage et la vérification. Dans un contexte agile, il est bon de spécifier que le design et le codage sont des activités conjointes. Le développeur est libre de choisir le design qu'il juge adéquat, afin de réaliser sa tâche. Il doit aussi tester son code avec des tests unitaires automatisés de préférence.

7 SCRUM

SCRUM est un processus de contrôle empirique de développement logiciel. Il permet aux équipes de produire des logiciels de manière itérative incrémentale, favorisant ce qui a le plus de valeur pour l'entreprise

SCRUM s'inspire de sources diversifiées. Elle est une amélioration du processus itératif et de l'approche incrémentale pour le logiciel orienté objet.

7.1 Vocabulaire de Scrum

Plusieurs éléments de SCRUM sont des métaphores référant au rugby. Le développement est perçu comme un jeu d'équipe. Selon Ken Schwaber « Les deux s'adaptent, sont rapides, s'organisent de manière autonome et ont peu de repos. ».

Backlog produit : Liste des éléments techniques et fonctionnels à produire dans un avenir prévisible pour le produit complet. Inclue ce qui est défini et ce qui reste à préciser.

Sprint backlog : Liste des éléments techniques et fonctionnels à produire au cours d'un sprint. C'est un sous-ensemble du backlog produit. Elle comporte des éléments suffisamment définis pour qu'ils soient réalisés au cours de la période du sprint.

Sprint : Une période de 30 jours où un ensemble de travaux sont effectués pour créer un livrable.

SCRUM : Rencontre journalière de l'équipe où l'avancement et les empêchements sont révisés. Il s'agit du plus fréquent et plus bas niveau de contrôle.

Règlement des rencontres SCRUM : Protocole des réunions quotidiennes pour être efficaces. Afin de minimiser la logistique, une série de règles uniformise la dynamique des rencontres.

Équipe SCRUM : Équipe multidisciplinaire qui travaille sur les éléments du sprint. Toutes les personnes (vente, service clientèle, etc.) impliquées dans le projet sont invitées à se joindre à l'équipe.

8 COMPARAISON ENTRE SCRUM ET CRYSTAL

Le tableau ci-dessous illustre une comparaison entre Scrum et Crystal sur quelques facteurs.

	Scrum	Crystal
<i>Documentation</i>	Ne met pas beaucoup l'accent sur la documentation, les documents minimum sont créés.	Ne met pas beaucoup l'accent sur la documentation, mais en raison de sa flexibilité, l'exigence d'une documentation fiable devient vitale.
<i>Implication de l'utilisateur final</i>	Les propriétaires du produit représentent les utilisateurs finaux.	Les utilisateurs participent à toutes les versions incrémentielles.
<i>Réunion de l'équipe</i>	Des réunions de stand up quotidiennes informelles ont lieu.	Des réunions face à face ont lieu.
<i>Taille des projets</i>	Peut être suivis pour des projet de toutes taille.	Peut être suivis pour des projets de toute taille en choisissant la bonne couleur.
<i>Rapidité</i>	Le projet est organisé autour des « sprints » de développement d'une durée allant généralement de deux à quatre semaines.	Les rythmes de développement et de livraison sont rapides (toutes les deux semaines ou une fois par mois) afin que les utilisateurs puissent passer les tests.
<i>Accueil du changement</i>	Scrum es plus discipliné. Par conséquent, une fois que le sprint démarre, il ne permet plus de modifications à l'intérieur de la portée du sprint.	Le Crystal est plus permissif. Par conséquent, il accepte les modifications selon les exigences et de la taille de l'équipe.
<i>Livraison</i>	Scrum livre selon la priorité des éléments du backlog.	Crystal délivre selon la criticité.
<i>Taille de l'équipe</i>	Scrum croit en « une méthode pour tous ». Quelle que soit la taille de l'équipe, une seule méthodologie s'applique ici. L'équipe globale se divise en équipes de mêlée. Par conséquent, le nombre d'équipes de mêlée peut augmenter, mais la taille de l'équipe de mêlée individuelle reste la même.	Crystal dispose de différentes méthodes pour chaque type de projet, en fonction du nombre de personnes qui travaillent dans l'équipe.

9 CONCLUSION

Les méthodes agiles de la famille Crystal fournissent des outils adaptés à quasiment tous les cas de figures, que ce soit en termes de nombre de fonctionnalités, de taille d'équipe et de projet. En fonction de la taille et de la complexité du projet, on pourra choisir une méthode plutôt légère ou au contraire une méthode plus complète.

Crystal fait ressortir les principes communs qui régissent les équipes gagnantes. Une méthode n'est qu'une série de conventions, qu'un groupe d'individu suivent. L'équipe est invitée à construire sa propre méthode sur la plate-forme que propose Crystal. La méthode qui en résulte recherche l'efficacité, doit être viable pour les membres de l'équipe et amène le projet dans une zone sécuritaire améliorant ainsi ses chances de succès.

10 REFERENCES

- ... à l'Ecole de la V.I.E ... agile et crystal. (2015, Novembre 18). Récupéré sur ... à l'Ecole de la V.I.E ...: <http://alecoledelavie.com/accueil/?p=1478>
- AgileThinkerNotes. (s.d.). *Note12: Crystal Clear, la méthode Agile des petites équipes*. Récupéré sur <https://agilethinkernotes.wordpress.com/2017/11/18/289/>
- Bénard, J.-L. (2001). *EXTREME PROGRAMMING - METHODES AGILES*. Récupéré sur www.businessinteractif.fr
- éditoriale, é. (s.d.). *Tout sur la méthode agile Crystal*. Récupéré sur nutcache: <https://www.nutcache.com/fr/blog/methode-agile-crystal/>
- FRICHET, t. (2016, septembre 03). *Crystal Clear : La méthode Agile des petites équipes méconnue en France*. Récupéré sur tfrichet.fr: <https://www.tfrichet.fr/crystal-clear-la-methode-agile-des-petites-equipes-meconnue-en-france/>
- Manale. (2019, août 21). *Découvrir la méthode Crystal Clear*. Récupéré sur Visual Planning: <https://www.visual-planning.com/fr/blog/decouvrir-la-methode-crystal-clear>
- Paquet, J. (2017, juin 29). *Qu'est-ce que le Crystal Clear ?* Récupéré sur my agile pattern: <https://blog.myagilepartner.fr/index.php/2017/06/29/quest-ce-que-le-crystal-clear/>
- TREMBLAY, R. (2007). *IMPLANTATION D'UNE MÉTHODE AGILE DE*. mémoire, FACULTÉ DES SCIENCES ET GÉNIE , Quebec.